

NASA Technical Memorandum 89038

A Method for Data Base Management and Analysis for Wind-Tunnel Data

Aileen O. Biser

MARCH 1987



A Method for Data Base Management and Analysis for Wind-Tunnel Data

Aileen O. Biser
Langley Research Center
Hampton, Virginia



National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1987

Contents

Summary	1
Introduction	1
Terminology	1
Discussion	2
The Problem and Previous Solution	2
The Search for a Solution	2
Solution Design	3
System Implementation	4
System Operation	5
Implementation by Other Organizations	5
Conclusions	6
References	6
Tables	7
Figures	11
Appendix A—Functional Requirement Specifications for the PAB Data Base Management System . .	19
Appendix B—Details of Data Base Schema	21
Appendix C—Menu System Execution	23
Appendix D—Execution of QUERY Program	25
Appendix E—Compressed RIM Data Base Creation	29

PRECEDING PAGE BLANK NOT FILMED

Summary

This paper describes the development of a method for the data base management and the subsequent analysis of wind-tunnel data. The design and implementation of the system are discussed and examples of its use are given. This system consists of a tree-structured menu as the main driver for access to applications and uses Relational Information Management (RIM) as the data base manager. The arrangement of the wind-tunnel data is described in the proper data base definition (schema) for efficient data base utilization, and recommendations for future use and improvements are given.

Introduction

Wind-tunnel investigations generate large amounts of data to support research projects. Following each investigation, the resulting data are analyzed, reported, and then relegated to some type of data storage. The amount of data generated during a year's period of time can be significant. For example, the Langley Propulsion Aerodynamics Branch (PAB) conducts about 20 programs per year in the Langley 16-Foot Transonic Tunnel and static test facility. In each test the data items generated average around 2.5 million, a result which means that about 50-million data items are stored for use in current and future research projects.

Because of the large amounts of data, it was determined that a well designed and managed data base and analysis system was needed to maximize the potential usefulness of the experimental data and to increase engineer productivity. The method that had been previously employed to perform these tasks was not considered adequate to handle this large and rapidly increasing complex task. The purpose of this project therefore has been to develop a system of efficient and user-friendly data management and analysis software that could fully meet the current and future needs of the experimental researcher with the capability for expansion and future development. In an effort to obtain further insight into the needs of the researcher, a survey was conducted of the various methods that are employed by individuals and groups to handle data management and analysis. In addition, a study was made of the nature of the data and an investigation was made of the functional requirements for an integrated system for data management and analysis.

These investigations led to the design of a data base management and analysis system that will be discussed in detail in this paper. In general, the system consists of a menu driver for access to data management, analysis, and utility programs. The data base management system selected for this project is the Relational Information Management (RIM) System described in reference 1. It offers several features, the most important of which are discussed in table I.

Having chosen a method as a solution to the problem, the implementation of this method as a system then became the major task. Preliminary pilot test cases for the system were set up in order to test the system. From this study, detailed information was gained on defining the optimum configuration for the entire system.

The resulting production system for the data management and analysis of experimental data at the Propulsion Aerodynamics Branch was installed in January 1985. Since that time, additional application programs have been installed. This paper will also present some initial information on what programs are available, the procedure for using the system, and insights that have been gained during the first year of operation.

Terminology

answer file	final product of any data reduction program
attribute	column in a relation
CP	static-pressure coefficient
DBMS	Data Base Management System
IPAD	Integrated Program for Aerospace Design
relation	two-dimensional table of data where columns represent attributes and rows represent data occurrences
RIM	Relational Information Management
row	one set of attribute values

schema	data base definition; a definition of the relations and their attributes that comprise the data base
SIFT	Standard Interface Tape

Discussion

The Problem and Previous Solution

The primary factor in deciding to look for a new approach for data management and analysis software was the consensus that the system being used was not adequate for present or long-term use. The system was functional and provided a means to accomplish some of the necessary tasks, but it had many disadvantages. Another contributing factor to the problem was the nature of the data and the variety of different types of data files. To begin with, the data that are generated from tests in the Langley 16-Foot Transonic Tunnel or the static test facility are passed through data reduction programs that produce a SIFT formatted answer file as shown in table II. The makeup of the NAME record on this file varies according to the nature of the test, the facility used for the test, and the version of the data reduction program used. The variety of the names and locations of items in the NAME record makes it difficult to set up application programs to access the data since these programs would constantly have to be modified in order to access the desired data. In addition, these data files are very large and contain a considerable amount of data that are not needed. The size of the data file causes data search and access time to be very slow.

The previous method used for data management was based on a program that produced a subset file of the data items that the researcher required for analysis. The file contained only data items that were pulled off the answer file. No other information was included on the subset and it was the user's responsibility to remember what items were included and their locations. The concept of dividing a large data file into small easily used subsets of needed data was sound, but the method of implementation was loosely organized. Figure 1 shows a diagram of the former method, in which the researcher would produce a subset file of the data from his master file and then use this subset as input to some application program. Depending on the makeup of this subset, he may or may not have to modify the application program to accept his subset file as input.

Before a data base is ready for final reporting and publication plots, the researcher must go through the process of creating a master file, which in turn is used to create a subset file that is then used in application programs. During this process, the researcher is responsible for the computer programming and manipulations necessary to complete these steps. These steps often have to be executed a number of times before accurate data are finally produced.

It is obvious that the researcher in many cases is acting as a computer programmer and operator as well as a researcher, which is his primary function. Much of his time is spent doing tasks that he should not have to concern himself with. The researcher who does not do his own computer operations and programming becomes dependent either on existing systems that may be slow or inadequate or on other researchers or contractors for help in producing data output, whether they be final data files, tabulated listings, or plots.

All these factors obviously hinder a researcher's productivity and lengthen the time between test completion and final report release. The final goal of this project was to define a method for data management and analysis that would provide the engineer easy and user-friendly access to the computer with a consistent and efficient means of producing final data. The extensive research required to accomplish this goal is described below.

The Search for a Solution

Accomplishing the goal of finding a consistent, efficient, user-friendly system required that specifications for a solution be researched and outlined. Once the functional requirements for a solution were decided upon, research was conducted into the possible methods of implementation.

Before formulating specifications for a solution, several researchers were interviewed to obtain their views on the current procedure and to acquire information on what they considered important features for a solution system to incorporate. The views expressed by the researchers pointed out the pros and cons of the current system; the primary ones are listed as follows:

Pros

1. Using small subset files provided faster and easier access to data items.
2. It is relatively simple to set up programs to access the data providing the researcher has some experience with programming.

Cons

1. The current system requires the researcher to have some programming skills and experience in working with computer commands and procedures.
2. It is difficult to delete, modify, or add data to an existing data base.
3. The development of programs was required to perform simple data base access operations.

The nature of the data and the process that the researcher went through to produce final data were also discussed. During these discussions common complaints and needs surfaced that were instrumental in the formulation of the functional requirement specifications for the design of a solution system. These functional requirements were then written down and discussed again to ensure that every need was included. These specifications, shown in appendix A, were used as guidelines for the design and ongoing development of the system.

Once the specifications were set down as guidelines for the design of a system, the next step became the search for a method of implementation that could satisfy the design specifications. The research began with a look at various methods of implementing a user-friendly facility and the major data base design techniques shown in references 2 and 3. The capabilities and tools available on the computer on which the system would be installed were studied. A survey was made of the various ways that experimental data management was handled by other groups and individuals. As a result of this research, three of the different data management methods found were considered in detail. These included the method used by the researchers in the PAB, a command-driven implementation used at another test facility, and a tree-structured implementation.

A method that met most of the specifications was found. This method used RIM as the data base manager in conjunction with a menu driver for access to application programs. This system was chosen as the basis on which the solution system was designed.

Solution Design

The primary solution design consists of a tree-structured menu for access to all applications and RIM-Version 6 (ref. 1), which was used as the data base manager. The primary reasons for choosing this design approach are ease of use and versatility. In addition, this approach represents the best way to meet the functional requirement specifications of the new system, as will be discussed in the following paragraphs.

The access point for all the data base management and application codes is the MENU program. Figure 2 illustrates the structure of the menu and the division of categories. The items included in this menu reflect the particular needs of the PAB at this time and not all categories are fully implemented. Those items proposed for implementation are indicated. The menu is designed so that categories can be added, deleted, or changed in any way at any given time. In this way the menu becomes a versatile tool in providing access to the many different application programs and utilities that are available. The need for the researcher to perform many computer operations is eliminated by having easy access through the menu and submenus to needed applications. This convenience solved some of the researcher problems, and the data management problems were to be solved by the RIM System (ref. 1).

RIM is a Relational Information Management System developed by The Boeing Computer Services Company under a NASA contract for IPAD. As was shown in table I, RIM has many important features that make it a good choice for a data base management system. It is versatile and can be applied to any type of data used by the Propulsion Aerodynamics Branch. Support for the RIM software and its users is provided by The Boeing Company. Additionally, it is important that RIM is available on some minicomputers because many of the tasks that are now performed at the main computer site will be performed on a minicomputer at some time in the future. The most important reason for choosing RIM as the data base manager was that it exceeded the required capabilities as outlined in the specifications. Because of its availability, a data management system utilizing RIM could be available sooner than developing new software.

Before going any further, more detailed information concerning RIM and the Data Base Management System (DBMS) should be understood. A data base is a computerized collection of stored data that serves the needs of multiple users within an organization. The DBMS is a generalized tool for manipulating large data bases; it is made available through special software for the interrogation, maintenance, and analysis of

data. The combination of DBMS software, applications software, data base implementation, and operating system/hardware environment brought together to provide information services for users is known as a data base system. The design of a system of this nature requires an understanding of the data and how it is used in order to provide the optimum implementation of the system.

Once RIM was chosen as the software for data management, a means had to be found for loading the data into correctly formatted files for the RIM software to access it. This process means that a schema for the data has to be decided upon and application programs be written to create RIM data files. The design of the schema is the most critical aspect of the system implementation.

In order to utilize fully the advantages of a data base management system, the schema for the data must represent functional divisions of the data. By using a relational approach to data management, these functional groups can be separated into different relations. Since wind-tunnel data have natural divisions, as shown in reference 4, these divisions were used in determining the makeup of relations. By using information gathered from the researchers concerning what they considered logical groupings of data items and also by considering the functional divisions of the data, a schema was developed as shown in appendix B. The modules as numbered in appendix B and the attributes are explained in detail in reference 4. The researchers considered these groupings to be the most useful method of arranging the data. In a relational data base it is normal to have attributes that are common to some or all relations in the data base. The purpose of this arrangement is to represent the dependencies between the data items and to serve as key attributes to identify the data rows uniquely. Table III shows what attributes are common to particular relations in the PAB schema. The purpose of having a consistent RIM schema format is to eliminate the wide variety of data file subsets that were being used. The names of the attributes have also been standardized in the RIM schema to eliminate problems dealing with variations in naming conventions on different data files. With consistent RIM formats and attribute names, the user need no longer rewrite his application programs to be compatible with each new data set.

In addition to the relations that have already been discussed, a relation called NAMS was also included in the data base schema. This relation contains information about the data base schema and can either be viewed interactively by the data base users or be accessed by the application programs to locate where particular attributes reside. This relation aids in writing applications to access a data base without needing to know the schema. Generalized application programs can be written to access data in any RIM data base that incorporates a NAMS relation as shown in table IV.

Once a design for the system had been decided upon and the schema developed, the problem of implementing the system was undertaken. The final production implementation is discussed below.

System Implementation

After deciding on the design of the system and the data base schema, the implementation of the system was begun. The resources used for this system are Control Data CYBER 700/800 Series computers available at the Langley Central Scientific Computer Complex. The first portion of the system to be installed was the menu. Appendix C demonstrates the execution of the menu and gives an outline of the items that are available. The MENU program is written in FORTRAN and serves as the driver for the applications that are available. The MENU was designed in such a way that a variety of different programs, specialized routines, and utilities could be added to the system or taken off easily at any time. In this way, many different programmers or researchers could add their programs as options on the system. Guidelines were set up for programmers to follow in order to incorporate their code into the system. These guidelines are discussed in the MENU system user's guide that is available as an item on the MENU system.

With the MENU system installed as the main driver of the system, a program to load experimental data into the RIM data files was needed. Because the system was required to be user friendly, an interface program was written to assist the user in executing the programs to load the RIM data file. This program called QUERY, shown in appendix D, allows the user to select interactively the relations he wishes to include in his data base. The user has the potential to create a data base that includes all the data on his answer file; but because most users need only a small portion of the data on the answer file, he can choose to include only those relations that contain the data items he needs. Appendix D gives an example of a typical interactive execution of QUERY. This program also contains a default creation mode, which eliminates the need for the user to specify relations, and a batch creation mode. More-detailed information concerning QUERY and its execution can be found in the QUERY user's guide that is available as an item on the MENU system.

Once the user has loaded his data into the RIM data file, he can then use these files as input to plotting and analysis application programs or, interactively, with the RIM stand-alone processor. The RIM user's guide

(ref. 1) explains the variety of commands available to the user for manipulating and viewing his data base. Data base users who are familiar with wind-tunnel data must only learn RIM commands to become proficient at interactive searching for data items or performing other data base functions.

System Operation

During the first year of active use, many problems associated with the design and implementation were encountered. The problems of excessive data file size and slow access time were not solvable because of hardware or software limitations, and in these cases the original specifications had to be modified. At the present time, the system incorporates a minimum of software needed to function usefully. For the complete data management and analysis system to be fully realized, additional software development of plotting and analysis programs is still required.

Using RIM and a menu facility has permitted the development of an organized and easily used wind-tunnel data base for experimental data that eliminates most of the previous problems. However, new problem areas have been created that should be addressed, and these include the following:

1. User education - Engineering user groups must be taught RIM commands and use of the FORTRAN interface library to make the necessary changes to existing application programs. The resistance to learning new methods hinders system development because important feedback from the users concerning system problems is not available.

2. Speed - After the system was in production for a time, it was discovered that certain portions were slow in execution because of large data base size and system overload. To alleviate this problem a new data base load program was designed and implemented to create a small specialized RIM data base for use with the applications that did not respond well with the large data bases. Appendix E discusses and shows a sample execution of this program.

3. Dependencies on host computer - Problems were encountered in maintaining a working system because of changes in the operating system and RIM. These changes were often announced and installed at the same time, a condition causing the data management system to be unavailable until the required revisions could be made. The efficiency of the system was hindered by being installed on overtaxed computer resources. Many of these problems could be alleviated by incorporating the system at a local level on a minicomputer or equivalent system capable of housing the data and software. In this way, prior knowledge of and control over system changes would be available and system resources would not be overtaxed.

4. Changes in data reduction - The system was designed to take data from the data acquisition system for the Langley 16-Foot Transonic Tunnel that had been converted for use by the data reduction program and then create a RIM data file. Any updates to the data reduction program that change the namelist on the answer file will have a direct effect on the conversion to a RIM data file. It is difficult to maintain a standardized and consistent schema, since changes due to different requirements for each test and the updating of the current data reduction system are continuously being made to the answer file namelist from which the schema was designed. Incorporating the data reduction process into the system may help alleviate the problem by providing control over where the necessary changes are being made in the answer file namelist.

5. Feasibility - The system consists of multiple programs and procedure files and has the potential for continued expansion; as a result, it is large and complex. Its functioning is dependent on many factors, such as changes in the namelist, operating system, and user's needs, as well as possible error conditions. The system requires continued maintenance, support, development, and enhancements, as well as consulting and training for users.

Implementation by Other Organizations

This data management system used by the PAB was designed for growth and expansion and possible widespread use. It is possible that this design structure could be applied to the data management problem of experimental data at other installations. In order to incorporate this system at another installation, slight modifications to the menu and schema may have to be made because of differences in the data collected. Tailoring the source code to load a data file into RIM is easily done; and by using this schema for RIM data base creation, no modifications to plotting and analysis programs would have to be made. In this way, duplicated effort in developing plotting and analysis programs could be reduced and researchers would have a wider range of tools to choose from. A method for handling wind-tunnel data that is consistent at all test facilities would be instrumental in the dissemination of the data. The design of the system acts as a basic

structure for others to use to build systems tailored to their needs, but systems closely related in nature so that the common use of applications and data could be accomplished easily.

If this system were to be used on a more widespread level, modifications and more detail concerning the implementation must be decided upon and added to the specifications. During implementation of the project, it was found that some changes in the data base design would be required to make the system more ideally suited to wind-tunnel data management. An initial effort in this direction was made with the design and implementation of the program shown in appendix E. Some of the problems that will need to be addressed include finding an easier way to modify multiple data items and a method to view items in different relations at the same time.

Conclusions

A data base management system was developed for the Langley Propulsion Aerodynamics Branch (PAB) for use in analyzing data obtained in the Langley 16-Foot Transonic Tunnel. As a result of this program, the following conclusions are given:

1. The system uses a tree-structured menu system to access data management, plotting, and analysis application programs. This structure gives the researcher easy user-friendly access to the applications he needs.
2. Relational Information Management (RIM) is used as the data base manager because it is versatile and easy to use. It provides the researcher with a way to view and manipulate his data as he wishes.
3. The system design is versatile and expandable. It can be modified for use at other test facilities and can be used as the framework for establishing a consistent method of data management for wind-tunnel data.

NASA Langley Research Center
Hampton, Virginia 23665-5225
December 11, 1986

References

1. *RIM-Version 6—Relational Information Management System and Report Writer User's Guide*. Central Scientific Computing Complex Document Z-3, NASA Langley Research Center, May 1985.
2. Teorey, Toby J.; and Fry, James P.: *Design of Database Structures*. Prentice-Hall, Inc., c.1982.
3. Date, C. J.: *An Introduction to Database Systems, Volume II*. Addison-Wesley Publ. Co., Inc., c.1983.
4. Mercer, Charles E.; Berrier, Bobby L.; Capone, Francis J.; Grayston, Alan M.; and Sherman, C. D.: *Computations for the 16-Foot Transonic Tunnel—NASA, Langley Research Center*. NASA TM-86319, 1984.

Table I. Features of RIM From Reference 1

BCS RIM, the Relational Information Management System developed by The Boeing Company, is a data base management system based on the relational algebra model for data management. Data are organized in tables that are referred to as relations. One or more relations make up the data base definition, or schema. Each table has both columns called attributes and data rows. Tables can be changed or combined using BCS RIM commands.

The system is available as a stand-alone system and through an application program interface. As a stand-alone system, BCS RIM may be executed in menu mode or command mode. The menu mode prompts the user for the input required to create and update the data base. The command mode is used for direct input of the desired BCS RIM commands.

Key features of BCS RIM include the following:

- Query language – A powerful, but easy-to-use, English-like command language. Data qualification is accomplished through use of a WHERE clause.
- Relational algebra – Relational commands for manipulating multiple relations.
- Schema modification – Commands to modify the schema definition without unloading the data base.
- RIM-to-RIM communication – An UNLOAD command to facilitate transfer of BCS RIM data bases from one machine to another.
- Plotting capabilities – A PLOT command that allows the user to plot up to 10 Y-variables against an X-variable.
- EDIT command – The capability to edit attribute values selectively in a manner similar to that of a system editor.
- Computed attributes – The capability to compute the value of an attribute, rather than displaying or loading a fixed value.
- ON-LINE HELP facility – Allows the user to get useful information about BCS RIM syntax and specific commands.
- MENU mode – A novice user can quickly define and load a BCS RIM data base through this method of interactive question and answer.
- Program interface – Allows the user to access BCS RIM through FORTRAN, COBOL, or PASCAL application programs.
- Expanded data types – BCS RIM's many data types include matrix and vector as well as standard data types. Data may also be of either fixed or variable length.
- Machine portable code – About 95 percent of the BCS RIM code is written in standard FORTRAN; this means that BCS RIM is available on a variety of computer systems.
- MicroRIM compatibility – The microcomputer version of RIM, MicroRIM, is a subset of BCS RIM. Data and data base schemas can be transferred between mainframe and microcomputers.
- Report writer option – This option provides the capability to create, modify, and delete report definitions for formatted output of the contents of one or more relations.

Table II. Example of a SIFT File

A SIFT file is the output from the data reduction of raw data collected in the Langley 16-Foot Transonic Tunnel. For example,

NAME	1791	POINT	TEST	RUN	CONFIG.	BATCH
UNIT	1791	NUMBER	NUMBER	NUMBER	NUMBER	NUMBER
DATA	1791	1.	398.	1.0	1.0	1.0
DATA	1791	2.	398.	1.0	1.0	1.0
DATA	1791	3.	398.	1.0	1.0	1.0
DATA	1791	4.	398.	1.0	1.0	1.0
DATA	1791	5.	398.	1.0	1.0	1.0
DATA	1791	6.	398.	1.0	1.0	1.0
DATA	1791	7.	398.	1.0	1.0	1.0

Each line represents a record. The first item on a record tells whether it is a NAME, UNIT, or DATA record. The second item on the record gives a count of the number of items that follow. Each of these example records has a total of 1793 items.

Table III. Key Attributes in Multiple Relations and Relations to Which They Belong
 [Relations are defined in appendix B]

ATTRIBUTES	RELATIONS
SEQ TEST RUN POINT CONFIG BATCH MACH ALPHA BETA PHI THETA PTJ/PO	ALL
PO QO PTO TTO	TPARMS INTJET PRIJET SECJET THRUST
FI	PRIJET PROPA PROPB
CO VRES CDTB FAPT PL RHO	PROPA PROPB

Table IV. Example NAMS Relation in a PAB RIM Data Base

RELNAM (a)	RLOC (b)	NAME (c)	UNIT (d)
TPARMS	1	TEST	NUMBER
TPARMS	2	RUN	NUMBER
TPARMS	3	POINT	NUMBER
TPARMS	4	CONFIG	NUMBER
TPARMS	5	BATCH	NUMBER
TPARMS	6	ALPHA	NUMBER
TPARMS	7	BETA	NUMBER
TPARMS	8	PHI	NUMBER
TPARMS	9	THETA	NUMBER
TPARMS	10	PTJ/PO	RATIO
TPARMS	11	PO	PSI
TPARMS	12	QO	PSI

^aRELNAM denotes the name of a relation in the data base.

^bRLOC denotes the location of the attribute in the relation.

^cNAME denotes the name of the attribute in the relation.

^dUNIT denotes the unit of the attribute in the relation.

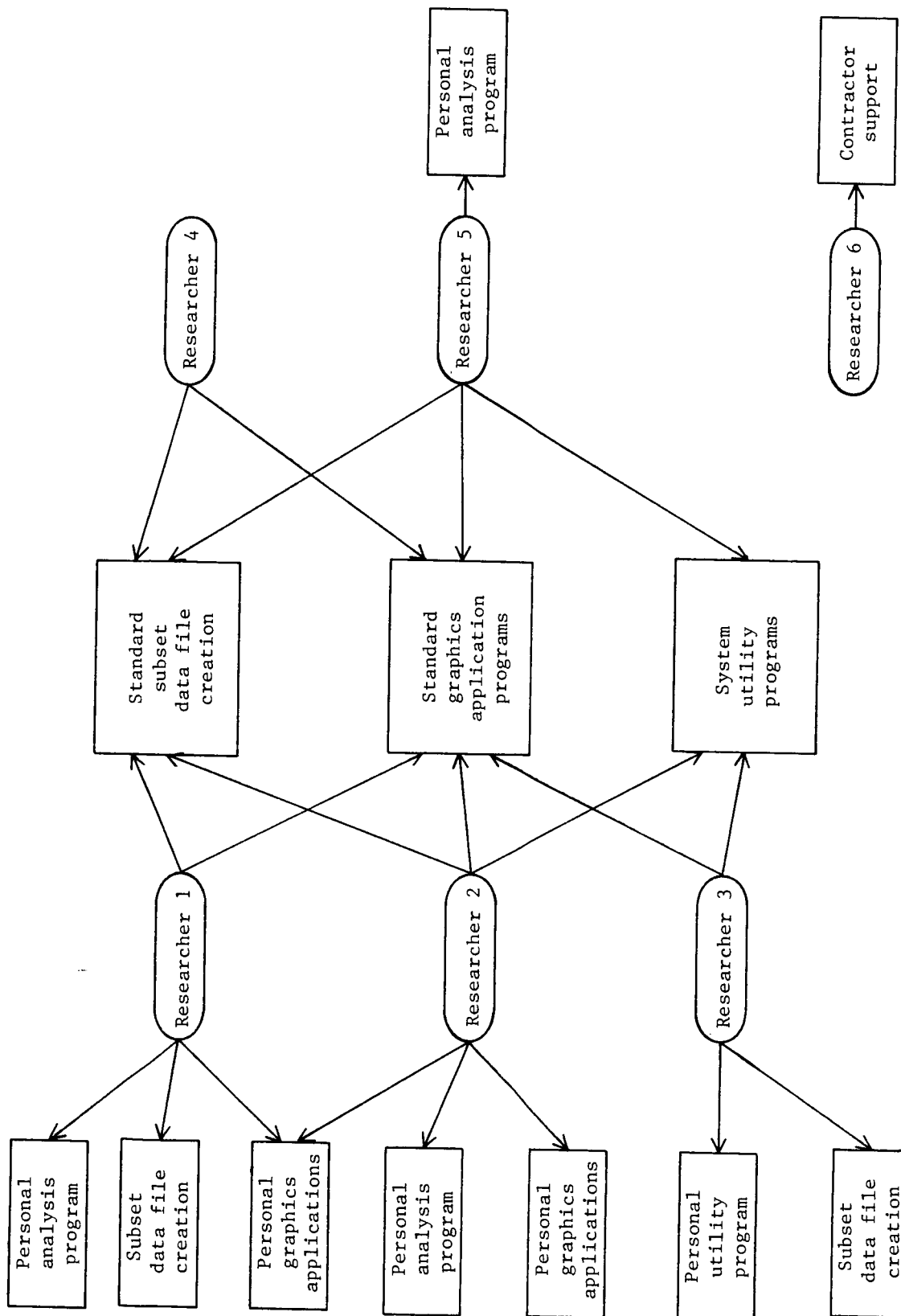
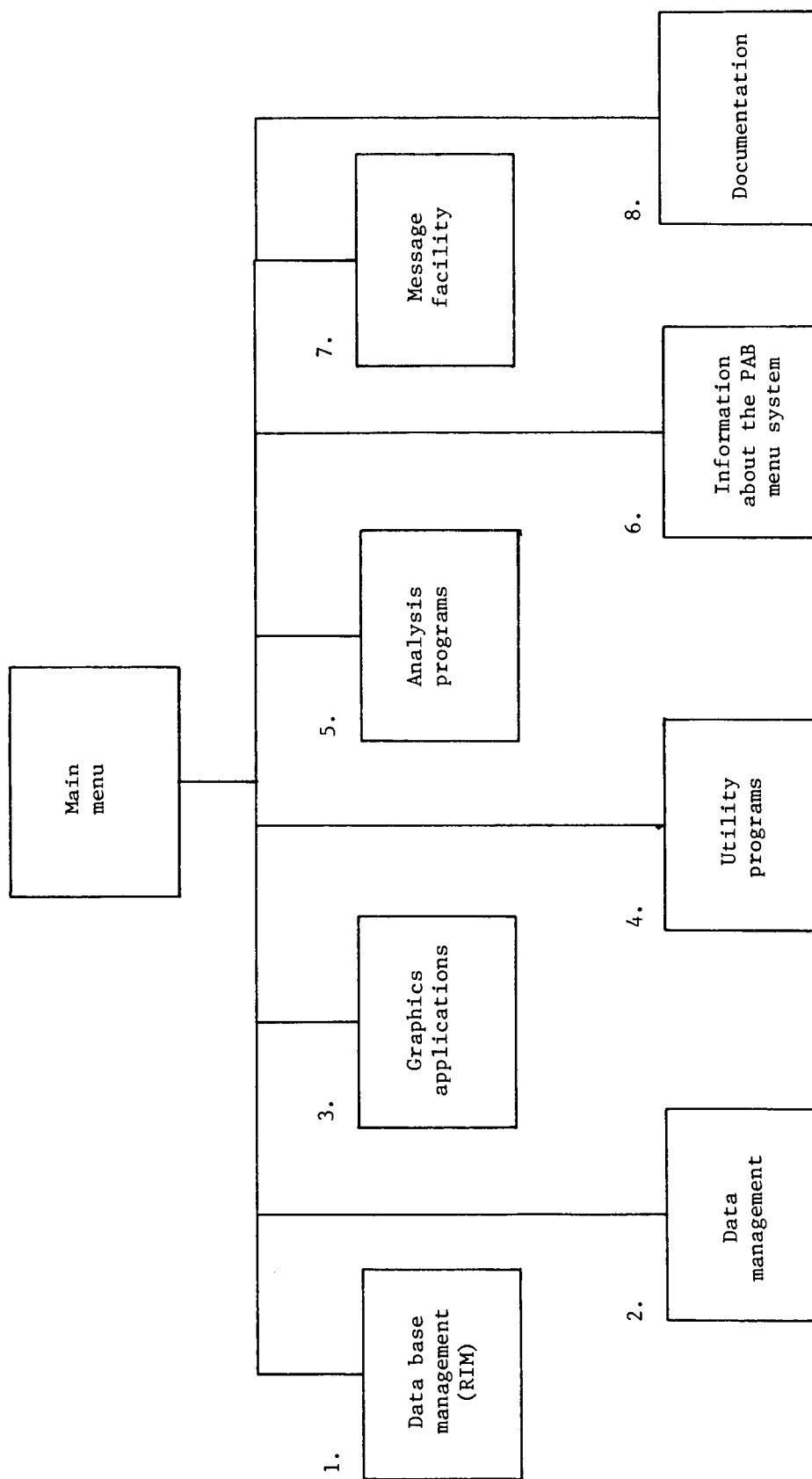
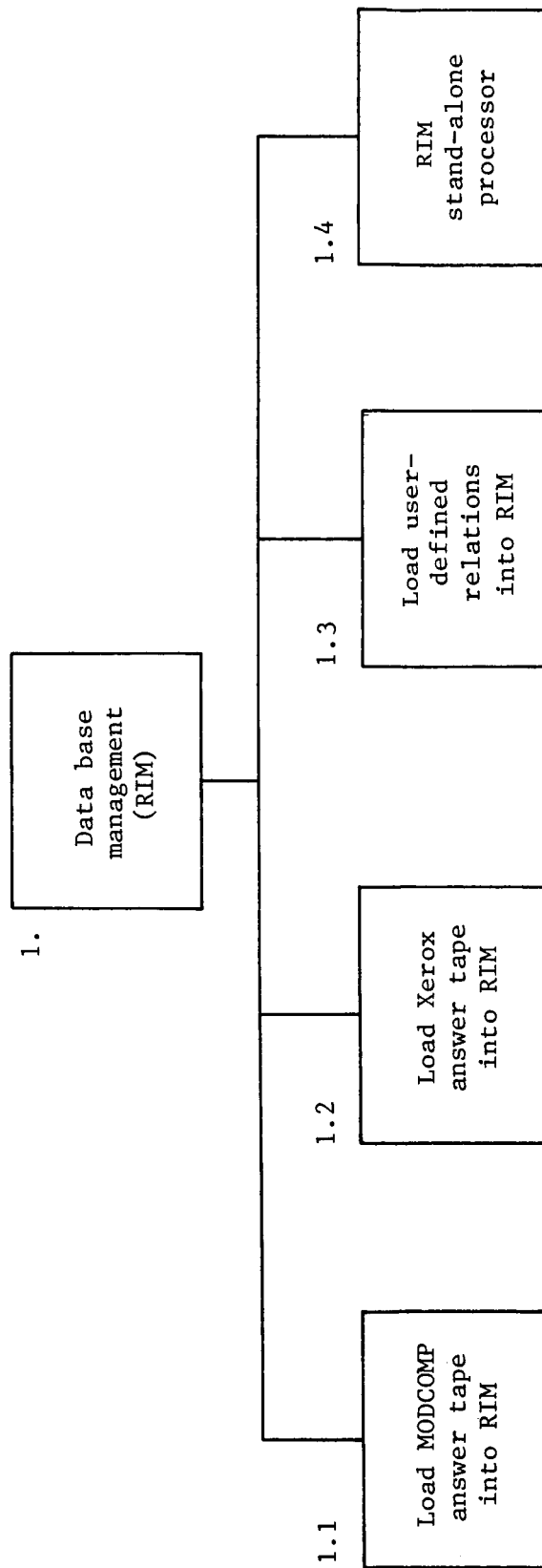


Figure 1. Former data base management method used at the PAB.



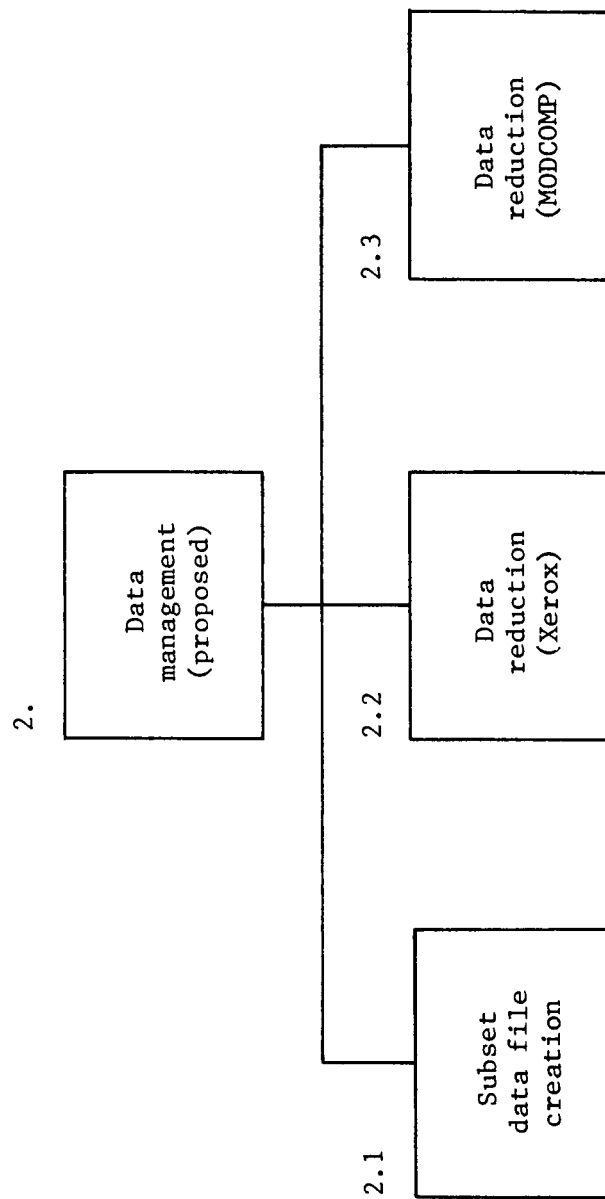
(a) Main menu.

Figure 2. MENU structure and content.



(b) RIM data base management submenu. MODCOMP is trademark of Modular Computer Systems, Inc.; Xerox is trademark of the Xerox Corporation.

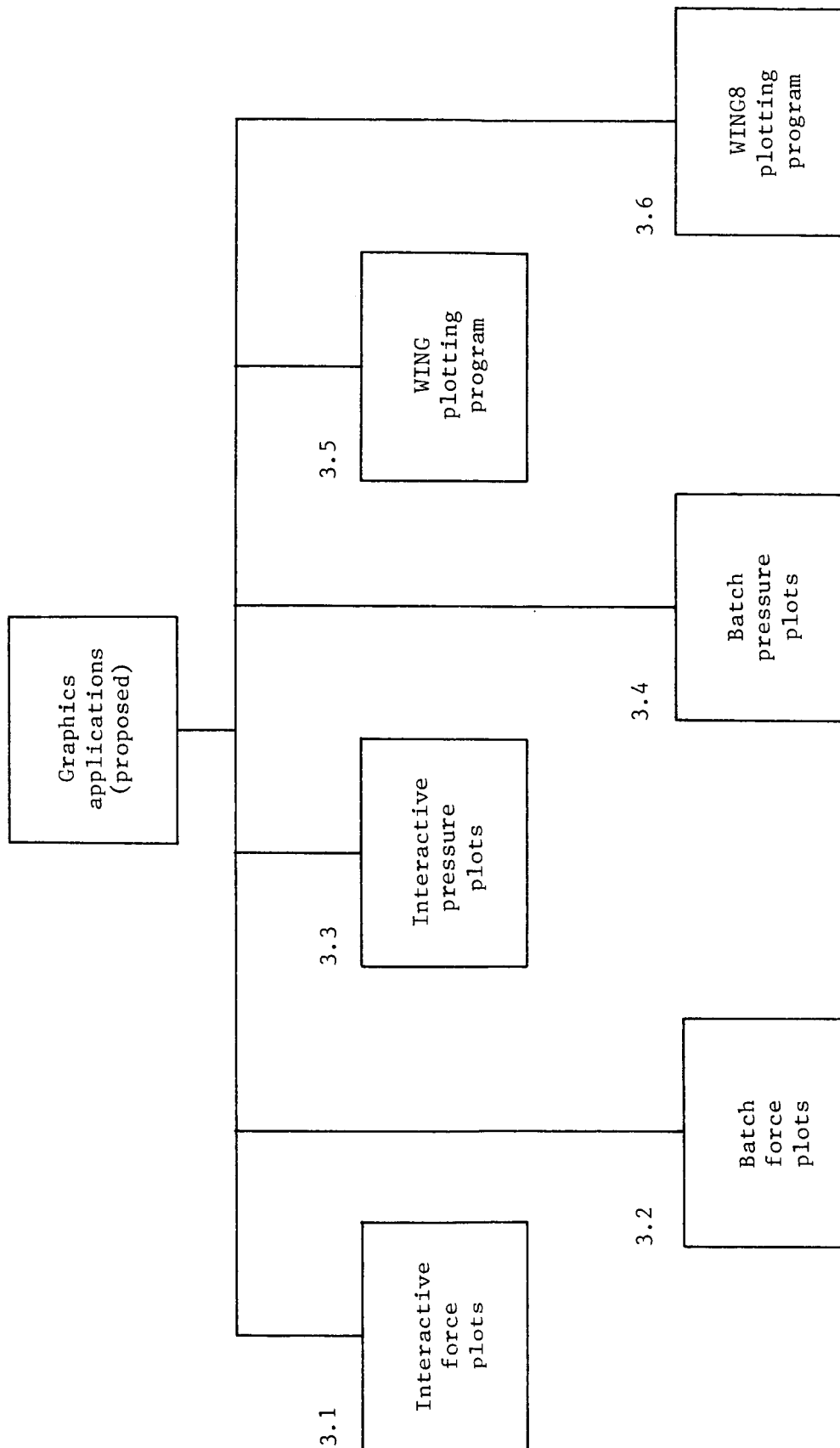
Figure 2. Continued.



(c) Data management submenu.

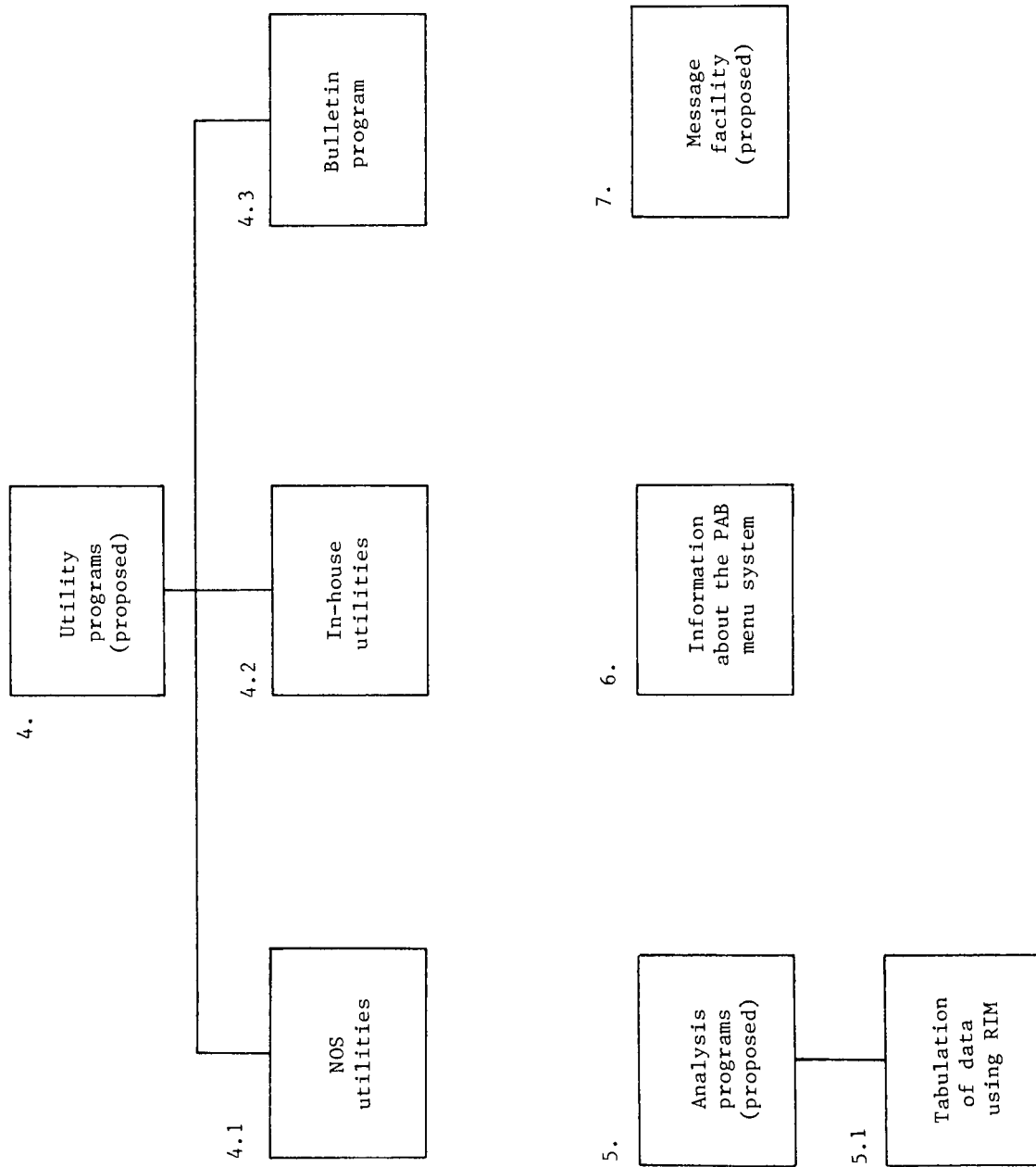
Figure 2. Continued.

3.

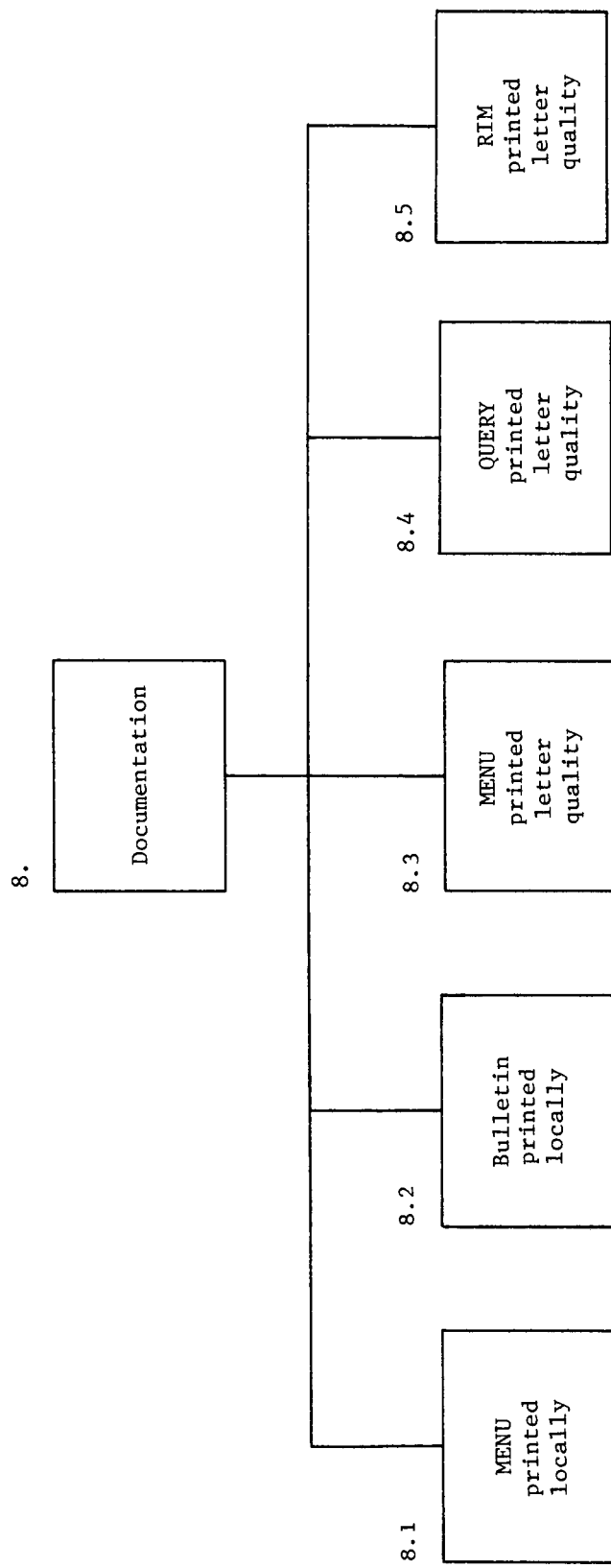


(d) Graphics applications submenu.

Figure 2. Continued.



(e) Utility programs, analysis programs, PAB information, and message submenus.
Figure 2. Continued.



(f) Documentation submenu.

Figure 2. Concluded.

Appendix A

Functional Requirement Specifications for the PAB Data Base Management System

The specifications outlined below are the original requirements decided upon in February 1984 to be used as design guidelines for the PAB Data Base Management System (DBMS).

Contents

- 1.0 Facilities
 - 1.1 Data base manager
 - 1.2 Documentation
 - 1.3 Interactive final-figure generator
 - 1.4 Batch final-figure generator
- 2.0 Performance targets
- 3.0 Reliability targets
- 4.0 Special requirements
- 5.0 Phasing

PRECEDING PAGE BLANK NOT FILMED

1.0 Facilities - functions the end user wishes the product to perform

1.1 Data base manager

1.1.1 Provide management of experimental or theoretical data

1.1.2 Provide capability to:

- Choose data items by name

- Make corrections to data

- Delete data

- Access data from a batch environment

- Select a subset of data for particular purposes

- Access data interactively

1.1.3 Accept SIFT formatted "answer file" as input

1.2 Documentation

1.2.1 Provide on-line help

1.2.2 User's guides

1.2.3 Programmer's guides

1.2.4 System documentation

1.3 Interactive final-figure generator

1.3.1 User friendly

1.3.2 Menu driven with user prompts

1.3.3 Provide final figures for:

- Pressure contours

- Single- and multiple-curve force and pressure plots

1.3.4 User-controlled plot design

1.3.5 Existing final-figure code interface

1.4 Batch final-figure generator

1.4.1 User friendly

1.4.2 Interactive input for batch execution

1.4.3 Provide final figures for:

- Pressure contours

- Single- and multiple-curve force and pressure plots

1.4.4 User-controlled plot design

1.4.5 Existing final-figure code interface

2.0 Performance targets - what is important to the user

2.1 Eliminate wait time at the terminal for interactive use and plot generation

2.2 Eliminate file manipulation by the user

2.3 Eliminate user's need-to-know operating system commands

3.0 Reliability targets

3.1 Minimum downtime for system failures

3.2 System changes thoroughly tested and users notified before implementation

4.0 Special requirements

4.1 Portability for future installation on a minicomputer

4.2 Data base output compatible with existing codes

4.3 Adaptable to future needs

5.0 Phasing

5.1 Initial pilot system release to gauge response to system

5.2 Incorporate improvements and release modified production system

5.3 Add additional facilities to system as they are developed and checked out

Appendix B

Details of Data Base Schema

This appendix shows the relations in a PAB data base and the data reduction modules (ref. 1) that each relation is drawn from. Additional details concerning the attributes in each relation can be obtained from the documentation available on the MENU system, which includes a complete list of the recommended standard attribute names.

Modules Related to Relations

Module		Relations	
I	Wind-tunnel parameters	- TPARMS	
II	Jet-exhaust measures	- PRIJET SECJET	
III	Skin-friction drag	- SKINFR	
IV	Balance loads and model attitudes	- BAL1 BAL2 BAL3 FORCECF	
V	Internal drag	- INTJET	
VI	CP and integration	- CPS-1 CPB-1 CPA-1 CPN-1 CPD-1 CPF-1 CPG-1 CPH-1	CPS-2 CPB-2 CPA-2 CPN-2 CPD-2 CPF-2 CPG-2 CPH-2
VII	Thrust removal	- THRUST	
VIII	Turboprop	- PROPA PROPB	

Appendix C

Menu System Execution

The PAB Executive is a menu-driven routine that allows the user to access various application programs, specialized routines, and utilities. The selection process begins with a display of the main menu. A selection is made by inputting the number corresponding to the desired choice. In this manner, the full range of selections is made available to the user.

This program displays a menu of programs or submenus from which the user interactively selects the function he wishes to execute.

To use the menu, enter the following commands:

1. `get,menu/un=059661n`
2. `-,menu`
or
`-,menu,help`
3. In response to the ? prompt, enter the digit corresponding to the selection you pick from the menu.

Other responses you may enter are:

- 0 - Return to main menu
- 20 - To quit (get out of menu)
- <CR> - To return to the next higher level menu (go to preceding menu)

This menu system will be used as a means of easily accessing and executing programs available within the Propulsion Aerodynamics Branch and on the Network Operating System (NOS). Its use is not restricted to the data management and analysis programs associated with RIM. Any other production-quality code may be added to the system.

The following examples show how the main menu and one of the submenus are currently set up. Because the menu program itself is dynamic, the choices shown and their order may change. These changes will come as the system grows and develops.

Example 1

MENU

- 1 DATA BASE MANAGEMENT (RIM)
- 2 DATA MANAGEMENT (OTHER)
- 3 GRAPHICS
- 4 UTILITIES
- 5 ANALYSIS
- 6 INFORMATION ABOUT THE PAB MENU
- 7 MESSAGES
- 8 DOCUMENTATION

- 0 - RETURN TO MAIN MENU
- 20 - QUIT
- <CR> - GO TO PRECEDING MENU

Example 2

DATA BASE MANAGEMENT (RIM)

- 1 LOAD MODCOMP ANSWER TAPE INTO RIM
- 2 LOAD XEROX ANSWER TAPE INTO RIM
- 3 LOAD USER-DEFINED RELATIONS INTO RIM
- 4 RIM

- 0 - RETURN TO MAIN MENU
- 20 - QUIT
- <CR> - GO TO PRECEDING MENU

Appendix D

Execution of QUERY Program

Procedure to Access QUERY

The wind-tunnel data from the Propulsion Aerodynamics Branch (PAB) are managed with the Relational Information Management (RIM) System. RIM is a data base management software package that organizes data into related tables of information. The data and related information are stored in three specially formatted files for use with RIM and application programs utilizing RIM.

The QUERY program assists the PAB experimentalists in creating the specially formatted RIM data files. QUERY takes an answer file and generates the RIM data base files after asking the user a series of questions aimed at tailoring the data base to his particular needs.

Shown below is one possible execution of the QUERY program. Additional examples are shown in the QUERY program user's guide.

The user can access the QUERY program for the PAB Executive MENU System. In order to run QUERY, one must first begin execution of the menu by typing:

```
get ,menu/un=059661n
-,menu
```

At this point you should choose item 1 as shown:

MENU

- 1 DATA BASE MANAGEMENT (RIM)
- 2 DATA MANAGEMENT (OTHER)
- 3 GRAPHICS
- 4 UTILITIES
- 5 ANALYSIS
- 6 INFORMATION ABOUT THE PAB MENU
- 7 MESSAGES

```
0 - RETURN TO MAIN MENU
20 - QUIT
<CR> - GO TO PRECEDING MENU
? 1
```

From the next menu choose item 1 as shown or item 2:

DATA BASE MANAGEMENT (RIM)

- 1 LOAD MODCOMP ANSWER TAPE INTO RIM
- 2 LOAD XEROX ANSWER TAPE INTO RIM
- 3 LOAD USER-DEFINED RELATIONS INTO RIM
- 4 RIM

```
0 - RETURN TO MAIN MENU
20 - QUIT
<CR> - GO TO PRECEDING MENU
? 1
```

This procedure will execute the QUERY program. Once you have loaded your data into RIM you can execute the RIM stand-alone processor (see instructions in the RIM user's guide—ref. 1) or any application program that accepts RIM data base files as input.

Example 1—Typical Interactive Data Base Creation

Interactive data base creation does not allow for any user-defined (USEDEF) relations (i.e., no engineering units). If you do choose interactive processing and also select USEDEF relations, the data base will be created,

but the USEDEF relations selected will not be included in the data base. The data base files will be created as you wait at the terminal during interactive execution. This example is of a force test with skin friction and thrust modules used.

ENTER A NAME FOR YOUR DATA BASE. (MAXIMUM OF 6 CHARACTERS)

? AT390B

YOUR PERMANENT DATA BASE FILES WILL BE:

AT390B1 AT390B2 AT390B3

ENTER THE PERMANENT FILE NAME OF YOUR DATA FILE.

? ANSWER

SELECT ANSWER FILE ORIGIN

1) MODCOMP

2) XEROX

? 1

SELECT PERMANENT FILE TYPE

1) INDIRECT ACCESS

2) DIRECT ACCESS

? 2

INPUT DELIVERY INFORMATION. EG. B1146 JOHN DOE

? B1146 BISER

DELIVERY INFORMATION IS: B1146 BISER.

IF INCORRECT, ENTER CORRECTION AT PROMPT

OTHERWISE ENTER CARRIAGE RETURN

?

<<<<<WARNING>>>>>

IF PASSWORD IS INCORRECT YOU WILL BE LOGGED OFF THE SYSTEM.

ENTER PASSWORD

?

DO YOU WISH TO CREATE A STANDARD DATA BASE? Y OR N

? N

SELECT JOB PROCESSING

1) INTERACTIVE (TABLES 1-30 VALID)

2) BATCH (ALL TABLES VALID)

? 1

THE AVAILABLE TABLE CHOICES ARE:

1. TPARMS	16. CPB-2	31. USEDEF1
2. BAL1	17. CPA-1	32. USEDEF2
3. BAL2	18. CPA-2	33. USEDEF3
4. BAL3	19. CPN-1	34. USEDEF4
5. FORCECF	20. CPN-2	35. USEDEF5
6. SKINFR	21. CPD-1	36. USEDEF6
7. INTJET	22. CPD-2	37. USEDEF7
8. PRIJET	23. CPF-1	38. USEDEF8
9. SECJET	24. CPF-2	39. USEDEF9
10. THRUST	25. CPG-1	40. USEDEF10
11. PROPA	26. CPG-2	41. USEDEF11
12. PROPB	27. CPH-1	42. USEDEF12
13. CPS-1	28. CPH-2	43. USEDEF13
14. CPS-2	29. MISC1	44. USEDEF14
15. CPB-1	30. MISC2	45. NAMS

ENTER NUMBERS TO SELECT RELATIONS.

INPUT NUMBERS IN FREE FORMAT. EG. 5 3 12 - 15 = 5 3 12 13 14 15

? 1 2 3 5 6 7 8 9 10

?

YOUR CHOICES ARE STARRED. (*)

* 1. TPARMS	16. CPB-2	31. USEDEF1
* 2. BAL1	17. CPA-1	32. USEDEF2
* 3. BAL2	18. CPA-2	33. USEDEF3
4. BAL3	19. CPN-1	34. USEDEF4
* 5. FORCECF	20. CPN-2	35. USEDEF5
* 6. SKINFR	21. CPD-1	36. USEDEF6
* 7. INTJET	22. CPD-2	37. USEDEF7
* 8. PRIJET	23. CPF-1	38. USEDEF8
* 9. SECJET	24. CPF-2	39. USEDEF9
* 10. THRUST	25. CPG-1	40. USEDEF10
11. PROPA	26. CPG-2	41. USEDEF11
12. PROPB	27. CPH-1	42. USEDEF12
13. CPS-1	28. CPH-2	43. USEDEF13
14. CPS-2	29. MISC1	44. USEDEF14
15. CPB-1	30. MISC2	45. NAMS

DO YOU WISH TO MAKE ANY CHANGES. Y OR N

? N

THE RELATIONS BEING LOADED ARE:

TPARMS	1	1
BAL1	2	2
BAL2	3	3
FORCECF	5	5
SKINFR	6	6
INTJET	7	7
PRIJET	8	8
SECJET	9	9
THRUST	10	1

10.10.13.BMAIO42

THE PROCFIL TO CREATE YOUR DATA BASE HAS
BEEN SUBMITTED. UPON COMPLETION, YOU
SHOULD FIND THESE DIRECT-ACCESS FILES IN
YOUR DIRECTORY:

AT390B1 AT390B2 AT390B3

REVERT. GOOD QPROC RUN

Appendix E

Compressed RIM Data Base Creation

Program Description

This program provides the user with an easy means to create a RIM data base from an answer file. This data base will be designed by the user and contain only the attributes and relations he requires. The user does this by supplying an input file to the program that contains the schema information. (See a following section, "Sample Input File.") The user must also supply a direct-access answer file and must provide a name to be used for the data base permanent-file creation. The data base name is limited to six alphanumeric characters and will be expanded with the suffixes 1, 2, or 3 by the program.

Access to Program

This program is available to users as option 3 under RIM data base management on the PAB menu system. (See a following section, "Sample Execution.") Once this job is initiated by the user, it can be detached for background execution or executed interactively.

Sample Input File

The sample input file is the file supplied by the user that defines the relations and attributes that will be in the RIM data base. In order for this data base to be used as input to WING, you must specify that the attributes 2, 3, and 6 in each relation be TEST, RUN, and BATCH, respectively, where TEST, RUN, and BATCH are locations 3, 4, and 6 on the namelist. The following example reflects this information:

	<u>Explanation</u>
2	(Number of relations)
REL1	(Relation name)
1 3 4 10 40	(Attribute locations from namelist)
6 120	
--EOR--	(EOF mark—use WEOF in Xedit)
--EOF--	
REL2	
5 3 4 44 59 6	
.	
.	
.	

Sample Execution

```
get,menu/un=059661n
-,menu
```

MENU

1	DATA BASE MANAGEMENT (RIM)
2	DATA MANAGEMENT (OTHER)
3	GRAPHICS
4	UTILITIES
5	ANALYSIS
6	INFORMATION ABOUT THE PAB MENU
7	MESSAGES
8	DOCUMENTATION
0	- RETURN TO MAIN MENU
20	- QUIT
<CR>	- GO TO PRECEDING MENU
? 1	

DATA BASE MANAGEMENT (RIM)

```
      1      LOAD MODCOMP ANSWER TAPE INTO RIM
      2      LOAD XEROX ANSWER TAPE INTO RIM
      3      LOAD USER-DEFINED RELATIONS INTO RIM
      4      RIM
0      - RETURN TO MAIN MENU
20     - QUIT
<CR>   - GO TO PRECEDING MENU
? 3
```

To create a user-defined data base:

```
Answer Tape ? afile
Input File Name? samin      NOTE: Entering a ? in
Data Base Name? samdb      response to these
STOP                       prompts will give
                           you on-line help.
```

DATA BASE MANAGEMENT (RIM)

```
      1      LOAD MODCOMP ANSWER TAPE INTO RIM
      2      LOAD XEROX ANSWER TAPE INTO RIM
      3      LOAD USER-DEFINED RELATIONS INTO RIM
      4      RIM
0      - RETURN TO MAIN MENU
20     - QUIT
<CR>   - GO TO PRECEDING MENU
? 20
REVERT. MENU TERMINATED NORMALLY
```

Sample Execution With Request for Help

To create a user-defined data base:

```
Answer Tape ? ?
ALLOWABLE VALUE(S)
MUST BE A FILE NAME
```

This parameter is the name of a direct-access file that is an answer tape file with NAME, UNIT, and DATA records.

```
Answer Tape ? afile
Input File Name ? ?
ALLOWABLE VALUE(S)
MUST BE A FILE NAME
```

This parameter is the name of an indirect-access file that contains the information for creating the data base schema. This includes the relation count, relation names, and the attribute locations. It should be set up as follows:

```

NUMBER OF RELATIONS
RELATION NAME
ATTRIBUTE LOCATIONS
-EOF-
RELATION NAME
ATTRIBUTE LOCATIONS
-EOF-

```

```

.
.
.

```

Attribute locations can be entered free format and on as many lines as needed. A series of locations can be specified with a dash (i.e., 20—30).

NOTE: In order to use this data base with WING, you must specify that TEST, RUN, and BATCH be in each relation in the order shown.

Attribute location		Answer tape location	
-----			To do this, input your choices in this order:
1			
2	TEST	3	___3 4___ ___6___...
3	RUN	4	
4			
5			
6	BATCH	6	
Input File Name? samin			
Data Base Name? ?			
ALLOWABLE VALUE(S)			

ANY 1—6 CHARACTERS FROM THE SET ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890

This parameter is the name you wish to use as your data base permanent file name.

NOTE: The data base internal file name will be PABDB to be consistent with all data bases.

Data Base Name? samdb

Sample Commands for Batch Execution

To execute this code as a batch job on D machine, use these commands in your batch stream:

```

GET,USERVAL.
GET,MENU/UN=059661N.
DFLINK,afile/M=W,NA,VF.
GET,samin.
BEGIN,NEWDB,MENU,afile,samin,samdb.
GET,xx1,xx2,xx3.
MFLINK,PABDB1,ST=CPF,I=xx1.
MFLINK,PABDB2,ST=CPF,I=xx2.
MFLINK,PABDB3,ST=CPF,I=xx3.

```

After these commands are executed, the files PABDB1, PABDB2, and PABDB3 are local to the job. The files—samdb1, 2, and 3—are permanent on D machine and must be MFLINKed back to the CPF system if

you wish to use them on A or Y machines. The files xx1, xx2, and xx3 can be any files you supply for the MFLINK command and should contain information that looks like this:

```
USER,acctno,password.  
CHARGE,chargeno,LRC.  
DEFINE,samdb_.
```

The blank in samdb_ indicates which data base file you are defining (i.e., samdb1, samdb2, or samdb3).

This is just one example of a method to set up your batch flow. You may know alternative methods. If you execute this job on A, C, or Y machines, you will not need to use the DFLINK and MFLINK commands.

Standard Bibliographic Page

1. Report No. NASA TM-89038	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle A Method for Data Base Management and Analysis for Wind-Tunnel Data		5. Report Date March 1987	
		6. Performing Organization Code 505-62-91-01	
7. Author(s) Aileen O. Biser		8. Performing Organization Report No. L-16198	
		10. Work Unit No.	
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225		11. Contract or Grant No.	
		13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001		14. Sponsoring Agency Code	
15. Supplementary Notes			
16. Abstract <p>To respond to the need for improved data base management and analysis capabilities for wind-tunnel data at the Langley 16-Foot Transonic Tunnel, research was conducted into current methods of managing wind-tunnel data and a method was developed as a solution to this need. This paper describes the development of the data base management and analysis method for wind-tunnel data. The design and implementation of the software system are discussed and examples of its use are shown.</p>			
17. Key Words (Suggested by Authors(s)) Data base Wind-tunnel data RIM Data analysis		18. Distribution Statement Unclassified—Unlimited	
		Subject Category 61	
19. Security Classif.(of this report) Unclassified	20. Security Classif.(of this page) Unclassified	21. No. of Pages 35	22. Price A03